Learning Simple External Contextual Languages from Positive Data

Leonor Becerra-Bonache

Learning SEC_p Languages from Only Positive Data*

Leonor Becerra-Bonache

Research Group on Mathematical Linguistics
Universitat Rovira i Virgili
Tarragona, Spain
E-mail: `leonor.becerra@estudiants.urv.cat`

Summary. The field of Grammatical Inference provides a good theoretical framework for investigating a learning process. Formal results in this field can be relevant to the question of first language acquisition. However, Grammatical Inference studies have been focused mainly on mathematical aspects, and have not exploited the linguistic relevance of their results. With this paper, we try to enrich Grammatical Inference studies with ideas from Linguistics. We propose a non-classical mechanism that has relevant linguistic and computational properties, and we study its learnability from positive data.

1 Introduction

Grammatical Inference (GI) is a subfield of Machine Learning that deals with the learning of formal languages. Roughly speaking, a GI problem can be defined as a gamed played between two players: a teacher and a learner. The teacher provides data to the learner, and from this data, the learner must identify the underlying language [4]. The initial theoretical foundations of GI

* This paper is based on [2] and [1].

were given by M.E. Gold [7], who was primarily motivated by the problem of first language acquisition. Since his seminal work, research in GI has focused on obtaining formal results (e.g, to find efficient methods for inferring grammars). Besides this theoretical bent, GI algorithms have also been applied to practical problems (e.g., Natural Language Processing, Computational Biology, etc.). Excellent surveys on the field of GI can be found in [6, 17].

Chomsky-inspired linguistic studies conceive grammar as a machine (in the sense of the theory of formal languages) that children develop and reconstruct very fast during the first years of their life. Children infer and select the grammar of their language from the data that the surrounding world offers them. Therefore, the proximity between GI and linguistic studies is considerable.

On the basis of these ideas, we try to bring together the theory of GI and studies of language acquisition, in pursuit of a final goal: to gain insight into the process of language acquisition. One concrete goal of this paper is to try to improve GI studies by using ideas from Linguistics. After presenting formal preliminaries (Section 2), we review the classes of languages on which GI studies have focused and we discuss whether they are suitable for modelling natural language syntax (Section 3). Then, we propose to study a non-classical mechanism that has important linguistic and computational properties (Section 4), and we study its learnability from positive data (Section 5). Concluding remarks and future work are presented in Section 6.

2 Preliminaries

In this paper we follow standard definitions and notations in formal language theory. Supplementary information can be found in [8].

Given an alphabet Σ, the set of all strings over the alphabet Σ is denoted by Σ^*. The set of nonempty strings from alphabet Σ is denoted Σ^+. A language L over Σ is a subset of Σ^*. The elements of L are called *strings* or *words*. λ is the *empty string*. Assume that $a \in \Sigma$ and $w \in \Sigma^*$; the length of w is denoted by $|w|$, and the number of occurrences of a in w is denoted by $|w|_a$.

$\mathbb{N}$ denotes the set of natural numbers. Assume that $\Sigma = \{a_1, a_2, ..., a_k\}$. The *Parikh mapping*, denoted by Ψ, is:

$$\Psi : \Sigma^* \rightarrow \mathbb{N}^k, \Psi(w) = (|w|_{a_1}, |w|_{a_2}, ..., |w|_{a_k})$$

If L is a language, then the *Parikh set* of L is defined by:

$$\Psi(L) = \{\Psi(w) \mid w \in L\}$$

A *linear set* is a set $M \subseteq \mathbb{N}^k$ such that $M = \{v_0 + \sum_{i=1}^{m} v_i x_i \mid x_i \in \mathbb{N}\}$, for some $v_0, v_1, ..., v_m$ in $\mathbb{N}^k$. A *semilinear set* is a finite union of linear sets, and a *semilinear language* is a language L such that $\Psi(L)$ is a semilinear set.

We denote by RE, CS, CF, LIN, and REG the families of languages generated by arbitrary, context-sensitive, context-free, linear, and regular grammars, respectively (RE stands for *recursively enumerable*). By FIN we denote the family of finite languages. The following strict inclusions hold: $FIN \subset REG \subset LIN \subset CF \subset CS \subset RE$. We call this, the *Chomsky hierarchy*.

3 Natural Languages and the Chomsky Hierarchy

GI studies have focused on learning REG and CF languages (i.e, the first two levels in the Chomsky Hierarchy) [6, 17]. However, the Chomsky Hierarchy has some limitations that should be taken into account when we want to study natural language syntax. One of the main limitations emerges when we try to locate natural languages in this hierarchy.

The question of determining the location of natural languages in the Chomsky Hierarchy has been a subject of discussion since it was posed by Chomsky in [3]. This debate focused on the following question: “Are natural languages CF?". However, in the late 80s, some clear examples of natural language structures that cannot be described using a context-free grammar were discovered (some examples of such constructions can be found in [11]). Linguists then agreed that natural languages are not CF.

It is worth noting that although the family of CF does not contain some important formal languages that appear in human languages, it has good computational properties. The family of context-sensitive languages contains all important constructions that occur in natural languages, but it is believed that the membership problem for languages in this family cannot be solved in deterministic polynomial time. Therefore, the question now is: “How much power beyond context-free is necessary to describe these non-context-free constructions that appear in natural language?"

The idea of generating CF and non-CF structures, and keeping the generative power under control, has led to the notion of *Mildly Context-Sensitive* (MCS), originally introduced by A.K. Joshi [9].

Definition 1. *By a Mildly Context-Sensitive family of languages we mean a family $\mathcal{L}$ of languages that satisfies the following conditions:*

(i) each language in $\mathcal{L}$ is semilinear,
(ii) for each language in $\mathcal{L}$ the membership problem is solvable in deterministic polynomial time,
(iii) $\mathcal{L}$ contains the following three non-context-free languages:
- *multiple agreements:* $L_1 = \{a^n b^n c^n \mid n \geq 0\}$
- *crossed agreements:* $L_2 = \{a^n b^m c^n d^m \mid n, m \geq 0\}$
- *duplication:* $L_3 = \{ww \mid w \in \{a, b\}^*\}$

The mechanisms for fabricating MCS families are well known (e.g., *tree adjoining grammars* ([10]), *head grammars* [16], *combinatory categorial grammars* [19], etc). All these studies are based on the idea that the class of natural languages is located in the Chomsky Hierarchy, between CF and CS (i.e., it includes REG and CF, but it is included in CS). However, as some authors have pointed out (for instance, see [12]), this assumption is not necessarily true, as natural languages could occupy an orthogonal position in the Chomsky Hierarchy (i.e., it contains some REG, some CF, etc.). In fact, we can find some constructions in natural languages that are neither REG or CF, and also some REG and CF constructions that do not appear naturally in sentences.

Taking these ideas into account, we consider that the study of natural language syntax from a formal point of view should focus on mechanisms that generate MCS languages and occupy an orthogonal position in the Chomsky Hierarchy. Unfortunately, most research on Grammatical Inference is not based on a class of languages with such features.

4 P-dimensional External Contextual Grammars

Contextual grammars were introduced by S. Marcus in [13], motivated by natural language investigations (for instance, modelling the acceptance of a word only in certain contexts). Roughly speaking, a contextual grammar produces a language starting from a finite set of words (*axioms*) and iteratively adding *contexts* (pair of words) to the currently generated words. Unlike the Chomsky grammars, contextual grammars do not involve nonterminals and they do not have rules of derivation except one general rule: to adjoin contexts. In the derivation process of the contextual grammars, the contexts can be added in two different ways: at the ends of the current string (these grammars are called *external*); or inside the current string (*internal* grammars).

Many variants have been investigated [15]. One of them is the so called *Many-dimensional External Contextual grammars.* These grammars extend the external contextual grammars, but work with vectors of words and vectors of contexts. Their linguistic relevance has been investigated in [11].

Let $p \geq 1$ be a fixed integer, and let Σ be an alphabet. A *p-word* x over Σ is a p-dimensional vector whose components are words over Σ, i.e., $x = (x_1, x_2, ..., x_p)$, where $x_i \in \Sigma^*, 1 \leq i \leq p$. A *p-context* c over Σ is a p-dimensional vector whose components are contexts over Σ, i.e., $c = [c_1, c_2, ..., c_p]$ where $c_i = (u_i, v_i), u_i, v_i \in \Sigma^*$, $1 \leq i \leq p$. We denote vectors of words with round brackets, and vectors of contexts with square brackets.

Definition 2. *Let $p \geq 1$ be an integer. A p-dimensional External Contextual grammar is $G = (\Sigma, B, C)$, where Σ is the alphabet of G, B is a finite set of p-words over Σ called the base of G, and C is a finite set of p-contexts over Σ. C is called the set of contexts of G.*

The direct derivation relation with respect to G is a binary relation between p-words over Σ, denoted by $\Rightarrow_G$, or $\Rightarrow$ if G is understood from the context. Let $x = (x_1, x_2, ..., x_p)$ and $y = (y_1, y_2, ..., y_p)$ be two p-words over Σ. By definition, $x \Rightarrow_G y$ iff $y = (u_1x_1v_1, u_2x_2v_2, ..., u_px_pv_p)$ for some p-context $c = [(u_1, v_1), (u_2, v_2), ..., (u_p, v_p)] \in C$. The derivation relation with respect to G, denoted by $\Rightarrow_G^*$, or $\Rightarrow^*$ if no confusion is possible, is the reflexive and transitive closure of $\Rightarrow_G$.

Definition 3. *Let $G = (\Sigma, B, C)$ be a p-dimensional External Contextual grammar. The language generated by G, denoted L(G), is defined as:*

$$L(G) = \{y \in \Sigma^* |\ \textit{there exists}\ (x_1, x_2, ..., x_p) \in B\ \textit{such that}\ (x_1, x_2, ..., x_p) \Rightarrow_G^* (y_1, y_2, ..., y_p)\ \textit{and}\ y = y_1y_2...y_p\}.$$

The family of all p-dimensional External Contextual languages is denoted by EC_p.

Remark 4.1 *Any family EC_p for $p \geq 2$ is a subfamily of linear simple matrix languages (see [11]).*

Definition 4. *A Linear Simple Matrix Grammar of degree n, $n \geq 1$, is a grammar $G = (N_1, ..., N_p, \Sigma, M, S)$, where:*

- *N_i: nonterminal alphabet.*

- *Σ: terminal alphabet.*
- *S: start symbol.*
- *M: finite set of matrices of the form*
  1. *$(S \rightarrow A_1 ... A_p)$, for $A_i \in N_i, 1 \leq i \leq p$, or*
  2. *$(A_1 \rightarrow x_1, A_2 \rightarrow x_2, ..., A_p \rightarrow x_p), for A_i \in N_i, x_i \in \Sigma^*, 1 \leq i \leq p$, or*
  3. *$(A_1 \rightarrow x_1 B_1 y_1, A_2 \rightarrow x_2 B_2 y_2, ..., A_p \rightarrow x_p B_p y_p)$, for $A_i, B_i \in N_i$, $x_i, y_i \in \Sigma^*, 1 \leq i \leq p$.*

Kudlek et al. [11] showed that for $p \geq 2$, the family EC_p is a MCS family of languages. They also showed that EC_p occupies an orthogonal position in the Chomsky Hierarchy. Therefore, EC_p is a mechanism with the desired properties described in Section 3.

5 The Simple p-Dimensional External Contextual Case

Taking into account the relevant properties of EC_p from a linguistic and computational point of view, in this section we will study its learnability from positive data.

One of the most important models investigated in GI is the model of *identification in the limit*, introduced by E.M. Gold in [7]. In this model, an infinite sequence of examples of the unknown language is presented to the learner, and its eventual or limiting behavior is used as the criterion of its success.

Definition 5. *Method M identifies language L in the limit if, after a finite number of examples, M makes a correct guess and does not alter its guess thereafter. A class of languages is identifiable in the limit if there is a method M such that given any language of the class and given any admissible example sequence for this language, M identifies the language in the limit.*

Two different learning settings are considered in this model: learning from text (only strings that belong to the language are given to the learner. It is also known as *learning from positive data*) and learning from informant (in addition to positive data, strings that do not belong to the language are also given to the learner).

Although it is desirable to learn from only positive data, Gold [7] proves that superfinite classes (i.e., classes of languages that contains all finite languages and at least one infinite language) are not identifiable in the limit from positive data. This implies that even the smallest class in the Chomsky Hierarchy (i.e., REG) is not identifiable in the limit from positive data.

According to the general definition, the EC_p grammar family is superfinite, since the base of G can be any finite set of p-words. We denote by p the dimension and by q the number of contexts.

Theorem 1. *The class EC_p is superfinite.*

Proof. Let $p = q = 1$. For any finite set S of strings over Σ, consider a EC_p with a base set S and an empty context set. Then, such a EC_p generates a finite language S. A EC_p with a base λ and a context set $\{[(a, \lambda)]\}$ can generate an infinite language a^*. Therefore, the language class is superfinite.

Corollary 5.1 *EC_p is not identifiable in the limit from positive data.*

Hence, we need to set some restrictions to make it possible to learn this class in the limit from only positive data.

Definition 6. *A Simple p-dimensional External Contextual grammar is $G = (\Sigma, B, C)$, where Σ is the alphabet of G, B is a singleton of p-words over Σ called the base of G, and C is a finite set of p-contexts over Σ. C is called the set of contexts of G.*

Therefore, a *Simple many-dimensional External Contextual* grammar is a subfamily of EC_p. The main difference is that the base of a Simple p-dimensional External Contextual grammar is restricted to a single p-word.

The family of all Simple p-dimensional External Contextual languages is denoted by SEC_p.

5.1 Properties of SEC_p grammars

Even if the base is a singleton, the family of SEC_p has several properties that are very interesting from a linguistic point of view. Here we present some of the most remarkable ones. On the basis of analogous arguments to those used by Kudlek et al. in [11], we can establish the following theorems.

Theorem 2. *For every integer $p \geq 2$, the family SEC_p is a MCS family of languages.*

Proof.
1. $SEC_p \subseteq EC_p$ and EC_p contains semilinear languages only (see [11]).

2. By *membership problem* the following is understood: given a language $L \subseteq \Sigma^*$ (defined by a certain type of grammar, automaton, etc.) and a word $w \in \Sigma^*$, decide algorithmically whether w is in L or not. Since the membership problem is polynomially decidable for EC_p, it follows that each family SEC_p, $p \geq 1$, is parsable in polynomial time (see [11], [14]).
3. The following languages are in SEC_p for every $p \geq 2$:
   - *multiple agreements*: $L_1 = \{a^n b^n c^n \mid n \geq 0\}$
   - *crossed agreements*: $L_2 = \{a^n b^m c^n d^m \mid n, m \geq 0\}$
   - *duplication*: $L_3 = \{ww \mid w \in \{a, b\}^*\}$

   It is easy to construct SEC_p grammars for each of these languages:

   (i) $L_1 = \{a^n b^n c^n \mid n \geq 0\}$. It is generated by the SEC_p grammar $G_1 = (\{a, b, c\}, B, C)$, where:

   - B $= \{(\lambda, \lambda)\}$
   - C $= \{\ c_1 = [(a, b), (c, \lambda)]\}$

   (ii) $L_2 = \{a^n b^m c^n d^m \mid n, m \geq 0\}$. It is generated by the SEC_p grammar $G_2 = (\{a, b, c, d\}, B, C)$, where:

   - B $= \{(\lambda, \lambda)\}$
   - C $= \{\ c_1 = [(a, \lambda), (c, \lambda)],\ c_2 = [(\lambda, b), (\lambda, d)]\}$

   (iii) $L_3 = \{ww \mid w \in \{a, b\}^*\}$. It is generated by the SEC_p grammar $G_3 = (\{a, b\}, B, C)$, where:

   - B $= \{(\lambda, \lambda)\}$
   - C $= \{\ c_1 = [(a, \lambda), (a, \lambda)],\ c_2 = [(b, \lambda), (b, \lambda)]\}$

Moreover, SEC_p occupies an orthogonal position in the Chomsky Hierarchy.

Theorem 3. *1. $SEC_p \subset CS$, for every $p \geq 1$.*
2. Each family SEC_p, $p \geq 2$, is incomparable with the family CF. The family SEC_1 is strictly contained in CF.
3. Each family SEC_p, $p \geq 1$, is incomparable with the family REG.

Proof. 1. Since no deletion is observed in the derivation process of a string in a SEC_p grammar, the first statement follows.

2. From Theorem 2 it follows that every family SEC_p, $p \geq 2$, contains noncontext-free languages. Consider now the context-free language $L = \{a^n b^n | n \geq 0\}^*$. Assume that L can be generated by a SEC_p grammar $G = (\Sigma, B, C)$. Consider the following word from L:

$$w = a^{i1} b^{i1} a^{i2} b^{i2} ... a^{ir} b^{ir},$$

   where $p < r$. One can easily see that by pumping all occurring contexts we cannot generate w , so L is not in SEC_p, for any $p \geq 2$.

   The second part of this statement follows from the fact that the family of external contextual languages is equal to MinLIN, which is a strict subfamily of LIN, incomparable with REG (see [11]).

3. Note that each family SEC_p, $p \geq 1$, contains nonregular languages. Now, consider the regular language $L = a^* \cup b^*$. One can verify that L is not in SEC_p, for any $p \geq 1$.

Figure 1 shows the location of SEC_p family in the Chomsky Hierarchy.

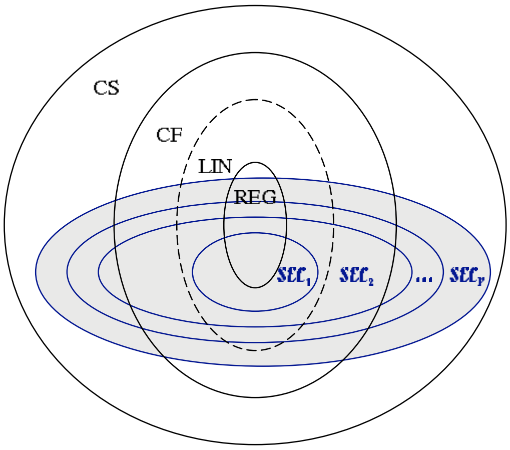

Fig. 1. The SEC_p family occupies an orthogonal position in the Chomsky hierarchy.

Moreover, the SEC_p grammar has another property with regard to EC_p grammars. We can find some languages showing the proper inclusion:

$$SEC_p \subset EC_p$$

For example, $L = \{a, b, c\}$. It is generated by an EC_p grammar, but can never be generated by a SEC_p grammar because of the restricted features of SEC_p grammars. This demonstrates that SEC_p is not superfinite.

5.2 Learnability of SEC_p languages from only positive data

Shinohara [18] showed that the class of languages generated by CS grammars with a fixed number of rules is learnable from only positive data. Hence, if we can transform a given SEC grammar with dimension *p* and degree *q* into an equivalent $LSMG$ (linear simple matrix grammar [5]) with dimension *p'* and degree *q'* and this into an equivalent CS grammar with a fixed number of rules, we will achieve our goal.

We will give the following constructive demonstration to prove that $SEC_{p,q} \subset LSMG_{p',q'} \subset CS$ grammars with a fixed number of rules.

First, we need to define *p*, *q*, *p'* and *q'*.

(i) $SEC_{p,q}$:
- p: dimension (in the same sense as SEC_p),
- q: degree (the number of contexts).

(ii) $LSMG_{p',q'}$:
- p': number of nonterminals in the right hand of the unique rule of the $LSMG$ started by S.
- q': number of matrices.

Let $G = (\Sigma, B, C)$ be a $SEC_{p,q}$ grammar, where

- B = $\{(\gamma_1, ..., \gamma_p)\}$
- C = { $c_1 = [(\alpha_1^1, \beta_1^1), ..., (\alpha_p^1, \beta_p^1)], ..., \ c_q = [(\alpha_1^q, \beta_1^q), ..., (\alpha_p^q, \beta_p^q)]$ }

We can transform this SEC grammar with dimension *p* and degree *q* into an equivalent $LSMG$ with dimension p' and degree q'.

G' = $(N_1, ..., N_p, \Sigma$, P, S), where

- P = { S $\longrightarrow A_1...A_p$,
$(A_1 \longrightarrow \gamma_1, ..., A_p \longrightarrow \gamma_p)$,
$(A_1 \longrightarrow \alpha_1^1 A_1 \beta_1^1, ..., A_p \longrightarrow \alpha_p^1 A_p \beta_p^1)$,
(...),
$(A_1 \longrightarrow \alpha_1^q A_1 \beta_1^q, ..., A_p \longrightarrow \alpha_p^q A_p \beta_p^q)$}

for $A_i \in N_i$, $\gamma_i, \alpha_i^j, \beta_i^j \in \Sigma^*$, $1 \leq i \leq p$, $1 \leq j \leq q$

The number of rules of an equivalent CSG will be proportional to p' · q'. Generally, there exists a CSG with the number of rules $\leq k \cdot p' \cdot q'$ (k is a constant).

We now illustrate this method using a grammar as follows. As a simple example, consider a $SEC_{p,q}$ with $p = 2$ and $q = 2$.

Let $G = (\{a, b, c, d\}, B, C)$ be a $SEC_{p,q}$ grammar, where

- B = $\{(ab, cd)\}$
- C = { c_1 = [(a, λ), (c, λ)], c_2 = [(λ, b), (λ, d)] }

Note that $L(G) = \{a^m b^n c^m d^n | m, n > 0\}$.

We can transform this SECgrammar with dimension p and degree q into an equivalent $LSMG$ with dimension p' and degree q'.

$G' = (\{S, A, A'\}, \{a, b, c, d\}, P, S)$, where

- P = { m_0: S $\longrightarrow$ AA',
 m_1: (A $\longrightarrow$ ab, A' $\longrightarrow$ cd),
 m_2: (A $\longrightarrow$ aA, A' $\longrightarrow$ cA'),
 m_3: (A $\longrightarrow$ Ab, A' $\longrightarrow$ A'd) }.

Now, we can construct a CSG: G" = (V_N, T, P', S), where

$V_N = \{S, A, A', B, R_1, R_2, R_3\}$

P' = {S $\longrightarrow$ ABA'

$AB \longrightarrow abR_1$	$R_1b \longrightarrow bR_1$	$R_1c \longrightarrow cR_1$	$bB \longrightarrow Bb$
$AB \longrightarrow aAR_2$	$R_2b \longrightarrow bR_2$	$R_2c \longrightarrow cR_2$	$cB \longrightarrow Bc$
$AB \longrightarrow AbR_3$	$R_3b \longrightarrow bR_3$	$R_3c \longrightarrow cR_3$	

$R_1A' \longrightarrow Bcd$	$R_1A' \longrightarrow cd$
$R_2A' \longrightarrow BcA'$	$R_2A' \longrightarrow cA'$
$R_3A' \longrightarrow BA'd$	$R_3A' \longrightarrow A'd\}$

Note that the set of rules presented here may contain some redundancy. However, we gave a priority to the consistency of the manner of constructing corresponding CSGs for general cases.

It is easy to prove that $L(G) = L(G') = L(G'')$. We will do it in two steps:

1. $L(G) \Leftrightarrow L(G')$

2. $L(G') \Leftrightarrow L(G'')$.

Proof 1

(i) $L(G) \Rightarrow L(G')$

Let $G = (\Sigma, B, C)$ be a $SEC_{p,q}$ grammar such that $L(G) = L$. Define the $LSMG_{p',q'}$ $G' = (N_1, ..., N_p, \Sigma, M, S)$ such that $A_i \in N_i, 1 \leq i \leq p$. The set M contains the following matrices:

- $(S \rightarrow A_1 A_2 ... A_p)$. The number of nonterminals in the right hand of the unique rule of the $LSMG_{p',q'}$ started by S, is equal to the dimension of the $SEC_{p,q}$. Therefore, p = p'.
- For the p-word $(x_1, x_2, ..., x_p)$, which constitutes the base of $SEC_{p,q}$, M contains the following matrix of rules: $(A_1 \rightarrow x_1, A_2 \rightarrow x_2, ..., A_p \rightarrow x_p)$. There is only one matrix of this kind because the base of the $SEC_{p,q}$ is a singleton (it has only one p-word).
- For each p-context $c = [(u_1, v_1), (u_2, v_2), ..., (u_p, v_p)] \in C$, M contains the matrix of rules: $(A_1 \rightarrow u_1 A_1 v_1, A_2 \rightarrow u_2 A_2 v_2, ..., A_p \rightarrow u_p A_p v_p)$. In this way, when we apply the contexts $c_1, c_2, ..., c_q$, we obtain the same result as when we apply the matrices $m_2, m_3, ..., m_{q+1}$, respectively.

It is easy to see that L(G') = L. By construction, for every $s \in L(G)$ there exists a derivation of s in G'.

(ii) $L(G) \Leftarrow L(G')$.

Let G' be the $LSMG_{p',q'}$, with $L(G') = L$. We define a $SEC_{p,q}$ grammar $G = (\Sigma, B, C)$ such that:

- For the matrix $(A_1 \rightarrow x_1, A_2 \rightarrow x_2, ..., A_p \rightarrow x_p) \in M$, B contains the p-word $(x_1, x_2, ..., x_p)$. Therefore, the elements of B coincide with the elements on the right hand of the matrix $(A_1 \rightarrow x_1, A_2 \rightarrow x_2, ..., A_p \rightarrow x_p)$.
- For each matrix of rules $(A_1 \rightarrow u_1 A_1 v_1, A_2 \rightarrow u_2 A_2 v_2, ..., A_p \rightarrow u_p A_p v_p) \in M$, the set C of p-contexts contains $c = [(u_1, v_1), (u_2, v_2), ..., (u_p, v_p)]$. Therefore, the number of matrices is equal to the number of contexts + 1.

It is easy to verify that L(G) = L. By construction, for every $s \in L(G')$ there exists a derivation of s in G.

Proof 2

(i) $L(G') \Rightarrow L(G'')$

Let $G' = (N_1, ..., N_p, \Sigma, M, S)$ be a $LSMG_{p',q'}$ such that L(G') = L. Define the CSG $G'' = (N, \Sigma, P, S)$, where: N is a finite set of nonterminal symbols, Σ is a finite set of terminal symbols that is disjoint from N, P is a finite set of production rules and $S \in N$ is the start symbol. The set P contains the following rules:

- $S \rightarrow A_1BA_2A_3...A_p$. The right hand of S coincides with the right hand of the unique rule started by S of the $LSMG_{p',q'}$. We add the nonterminal B when $p \geq 2$, to allow applications of different rules.
- For each matrix of M, P contains the following rules:
  - For the first rule of each matrix, P contains:
    $A_1B \rightarrow x_1R_1$
    $A_1B \rightarrow u_1A_1v_1R_2$
    (...)
    $A_1B \rightarrow u_1A_1v_1R_{q'}$
    q' is the number of matrices. So, there are correspondences between choosing the rule that contains R_1, for example, and applying matrix m_1.
  - For the second rule of each matrix, P contains:
    $R_1A_2 \rightarrow x_2R_1$
    $R_2A_2 \rightarrow u_2A_2v_2R_2$
    (...)
    $R_{q'}A_2 \rightarrow u_2A_2v_2R_{q'}$
    We apply this kind of rule from the second to the $p-1$ rule of each matrix (note that each matrix has p rules).
  - For the p rule of each matrix, P contains:
    $R_1A_p \rightarrow Bx_p \mid x_p$
    $R_2A_p \rightarrow Bu_pA_pv_p \mid u_pA_pv_p$
    (...)
    $R_{q'}A_p \rightarrow Bu_pA_pv_p \mid u_pA_pv_p.$
    If we use the rule that contains the nonterminal B, we will go back and apply more rules. Otherwise, we will finish the derivation.
  - We will need to add some intermediate rules to allow us to make the necessary derivations. These rules don't have any correspondence with the $LSMG_{p',q'}$. With these intermediate rules, we swap R_i to the right until it is adjacent to an A_i, allowing us to apply another

rule. Similarly, we move B to the left until it is adjacent to A_1, and then start to apply this process again.

It is easy to see that $L(G'') = L$. By construction, for every $s \in L(G')$ there exists a derivation of s in G''.

(ii) $L(G') \Leftarrow L(G'')$

Let G" be the CSG, with L(G") = L. We define a $LSMG_{p',q'}$ $G' = (N_1, ..., N_p, \Sigma, M, S)$ such that:

- For the unique rule started by S of the CSG, M contains the same rule without the nonterminal B.
- For all the rules that contain R_i in the CSG (except intermediate rules), where $1 \leq i \leq q'$, M contains a matrix with all these rules, but B, R_i and repeated rules are deleted.

It is easy to verify that L(G') = L. By construction, for every $s \in L(G'')$ there exists a derivation of s in G'.

Hence, there are clear relationships between $SEC_{p,q}$, $LSMG_{p',q'}$ and CSG.

(i) $p' = p$ (in our example, p is equal to 2; therefore, the number of nonterminals in the right hand of the unique rule of the $LSMG$ started by S is 2).
(ii) $q' = q + 1$ (in our example, q is equal to 2; therefore, the number of matrices of $LSMG$ has to be 3).
(iii) *The fixed number of rules of CSG is proportional to p'·q'.* Generally, one can have G'' with $O(p' \cdot q')$ number of rules. Since p' and q' are given, G'' has a bounded number of rules.

From a result by Shinohara [18], we can obtain the following theorem:

Theorem 4. *Given $p' > 0$ and $q' > 0$, the class of languages generated by linear simple matrix grammars with dimension p' and degree q' is learnable from positive data.*

Corollary 5.2 *Given $p > 0$ and $q > 0$, the class of languages generated by simple external contextual grammars with dimension p and degree q is learnable from positive data.*

Although what we have proved is enough to show that SEC can be learned from only positive data, we have a stronger result. As we will prove below, SEC with any dimension, but with at most q contexts and m bases, has finite elasticity (a sufficient condition for learning from positive data).

We will use the notation $\subset$ to mean a *proper* subset relation in the sequel.

By $\mathcal{S}ec(p, q, m)$, we denote the class of languages that can be generated by $SECs$ with a dimension that is less than or equal to p, with at most q contexts, and with at most m bases. By $\mathcal{S}ec(*, q, m)$, we denote the class of languages defined by

$$\mathcal{S}ec(*, q, m) = \bigcup_{p=1}^{\infty} \mathcal{S}ec(p, q, m).$$

Let w be a string over Σ. A pair (b, C) of a base b and a set C of contexts is said to *minimally generate* w if and only if w is generated by using a base b and contexts in C and there exists no b' and C' such that $b = b'$, $C' \subset C$ and w is generated by using b' and C'. For a string w, by $MinC(w)$, we denote the set of all pairs (b, C) (b:base, C:set of contexts) which minimally generate w. It is clear that the following lemma holds:

Lemma 1. For any $w \in \Sigma^*$, $MinC(w)$ is finite.

Theorem 5. *The class $\mathcal{S}ec(*, q, m)$ has finite elasticity. Therefore, it is identifiable in the limit from positive data.*

Proof. Assume that the class $\mathcal{S}ec(*, q, m)$ has infinite elasticity.

There exists an infinite sequence $w_0, w_1, w_2, ...$ of strings in Σ^* and an infinite sequence $L_1, L_2, ...$ of languages in $\mathcal{S}ec(*, q, m)$ such that, for any $k \geq 1$, $\{w_0, w_1, ..., w_{k-1}\} \subseteq L_k$ and $w_k \notin L_k$ hold.

For each $i = 1, 2, ...$, let S_i be some SEC generating L_i. Note that each S_i includes some element of $MinC(w_0)$ in its base and context set. Since $MinC(w_0)$ is finite by the above lemma, there exists $C_0 \in MinC(w_0)$ such that infinitely many S_i's include C_0. Let $\sigma = S_{n_1}, S_{n_2}, ...$ be an infinite sequence of such SEC's including C_0. Note that σ is a subsequence of $S_1, S_2, ...$ (That is $n_1, n_2, ...$ is a subsequence of 1,2,3,....)

The string w_{n_1} is not an element of L_{n_1}. Therefore, w_{n_1} is not generated by S_{n_1}. But, the infinite subsequence $S_{n_2}, S_{n_3}, S_{n_4}, ...$ should generate w_{n_1}, and therefore, should include some element of $MinC(w_{n_1})$ in its base and context set. Since $MinC(w_{n_1})$ is finite, there exists $C_1 \in MinC(w_{n_1})$ such that infinitely many S_{n_j}'s include C_1. Note that C_0 does not generate w_{n_1}. Therefore, $|C_0| < |C_0 \cup C_1|$ holds.

Repeating the same discussion, we can find an infinite sequence $C_0, C_1, \ldots$ satisfying the following conditions:

1. $|C_0| < |C_0 \cup C_1| < |C_0 \cup C_1 \cup C_2| < \cdots$ holds,
2. for any q, there exist infinitely many SEC's in $S_1, S_2, \ldots$ which include $C_0 \cup \cdots \cup C_q$ as its base and context set.

These conditions contradict the fact that the number of contexts and bases are upper bounded by q and m, respectively. This completes the proof.

6 Concluding Remarks

Despite the fact that REG and CF grammars are mechanisms with limited representational power to describe some constructions that appear in natural languages, GI studies have focused on them. In this paper we have proposed to study classes of languages that are more relevant from a linguistic point of view.

On one hand, we have seen that MCS languages provide a grammatical environment for natural language constructions. On the other hand, we have given some arguments that support the idea that natural languages could occupy an orthogonal position in the Chomsky Hierarchy. Therefore, it would be very interesting to study mechanisms with these properties (i.e., they fabricate MCS languages and they occupy an orthogonal position in the Chomsky Hierarchy).

P-dimensional External Contextual grammars are an example of a mechanism with such features. Hence, we believe they could have a chance in the study of natural language syntax. However, in order to study its learnability from only positive data, we have to restrict the grammar. So, we have introduced a new class of languages called Simple External Contextual. We have shown that this class with fixed dimension and degree is learnable from positive data, from Shinohara's results [18]. Moreover, we have presented a second stronger result that shows that Simple External Contextual with any dimension, but at most q contexts and m bases, has finite elasticity (sufficient condition for positive data learnability).

In the future, we would like to have a better understanding of the properties of the new class proposed and extend these learnability results. Moreover, taking into account that corrections are also available to the child in the early stages of language acquisition, and that the idea of corrections has been successfully applied to learn REG languages [1], we would like to study the learnability of Simple External Contextual languages using positive data and corrections.

References

1. Becerra-Bonache, L. (2006). *On the Learnability of Mildly Context-Sensitive Languages using Positive Data and Correction Queries*. PhD thesis, Rovira i Virgili University, Tarragona.
2. Becerra-Bonache, L. and T. Yokomori (2004). Learning mild context-sensitiveness: Toward understanding children's language learning. In *Proceedings of ICGI'04*. Berlin: Springer.
3. Chomsky, N. (1956). Three models for the description of language. *IRE Transactions on Information Theory*, 3: 113–124.
4. Clark, A. (2004). Grammatical inference and first language acquisition. In *Proceedings of Workshop on Psycho-Computational Models of Human Language Acquisition*, pp. 25–32.
5. Dassow, J. and Gh. Păun (1989). *Regulated Rewriting in Formal Language Theory*. Berlin: Springer.
6. de la Higuera, C. (2005). A bibliographical study of grammatical inference.*Pattern Recognition*, 38: 1332–1348.
7. Gold, E.M. (1967). Language identification in the limit. *Information and Control*, 10: 447–474.
8. Hopcroft, J.E., R. Motwani and J. D. Ullman (2001). *Introduction to Automata Theory, Languages and Computation*. Addison-Wesley.
9. Joshi, A.K. (1985). How much context-sensitivity is required to provide reasonable structural descriptions: Tree adjoining grammars. In D. Dowty, L. Karttunen and A. Zwicky (eds.), *Natural Language Parsing: Psychological, Computational and Theoretical Perspectives*, pp. 206–250. New York: Cambridge University Press.
10. Joshi, A.K. and Y. Schabes (1997). Tree-adjoining grammars. In G. Rozenberg and A. Salomaa (eds.), *Handbook of Formal Languages*, volume 3, pp. 69–123. Berlin: Springer.
11. Kudlek, M., C. Martín-Vide, A. Mateescu and V. Mitrana (2002). Contexts and the concept of mild context-sensitivity. *Linguistics and Philosophy*, 26(6): 703–725.
12. Manaster-Ramer, A. (1999). Some uses and abuses of mathematics in linguistics. In C. Martín-Vide (ed.), *Issues in Mathematical Linguistics*, pp. 73–130. Amsterdam: John Benjamins.

13. Marcus, S. (1969). Contextual grammars.*Revue Roumaine des Mathématiques Pures et Appliquées*, 14: 1525–1534.
14. Mitrana, V. (2005). Marcus external contextual grammars: From one to many dimensions.*Fundamenta Informaticae*, 54: 307–316.
15. Păun, Gh. (1997). *Marcus Contextual Grammars*. Dordrecht: Kluwer.
16. Roach, K. (1987). Formal porperties of head grammars. In A. Manaster-Ramer (ed.), *Mathematics of Language*, pp. 293–348. Amsterdam: John Benjamins.
17. Sakakibara, Y. (1997). Recent advances of grammatical inference. *Theoretical Computer Science*, 185: 15–45.
18. Shinohara, T. (1994). Rich classes inferable from positive data: Length-bounded elementary formal systems. *Information and Computation*, 108: 175–186.
19. Steedman, M. (1985). Dependency and coordination in the grammar of Dutch and English. *Language*, 61: 523–568.